

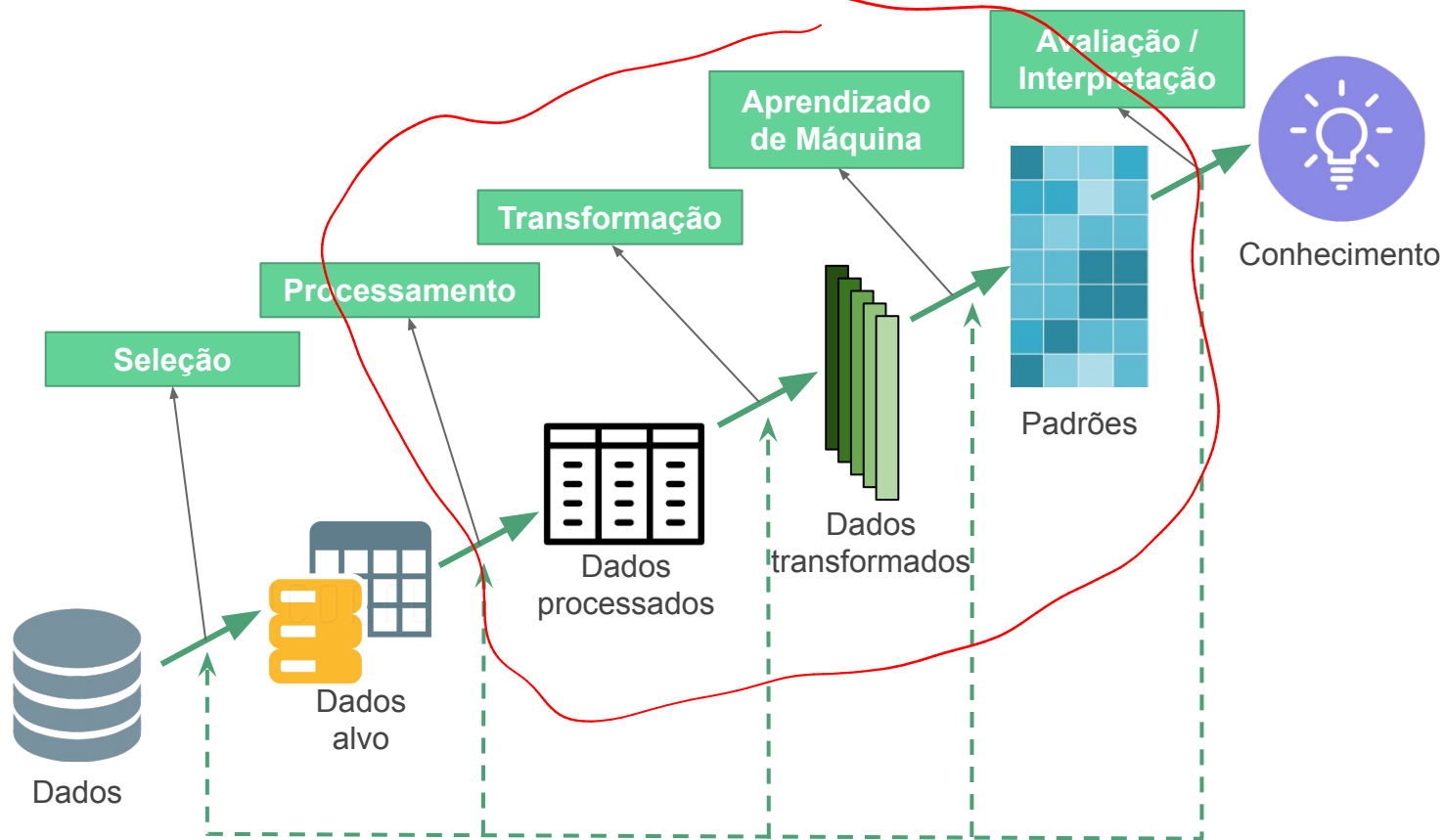
# Análise de séries temporais

por similaridade e alinhamento não  
linear com Dynamic Time Warping

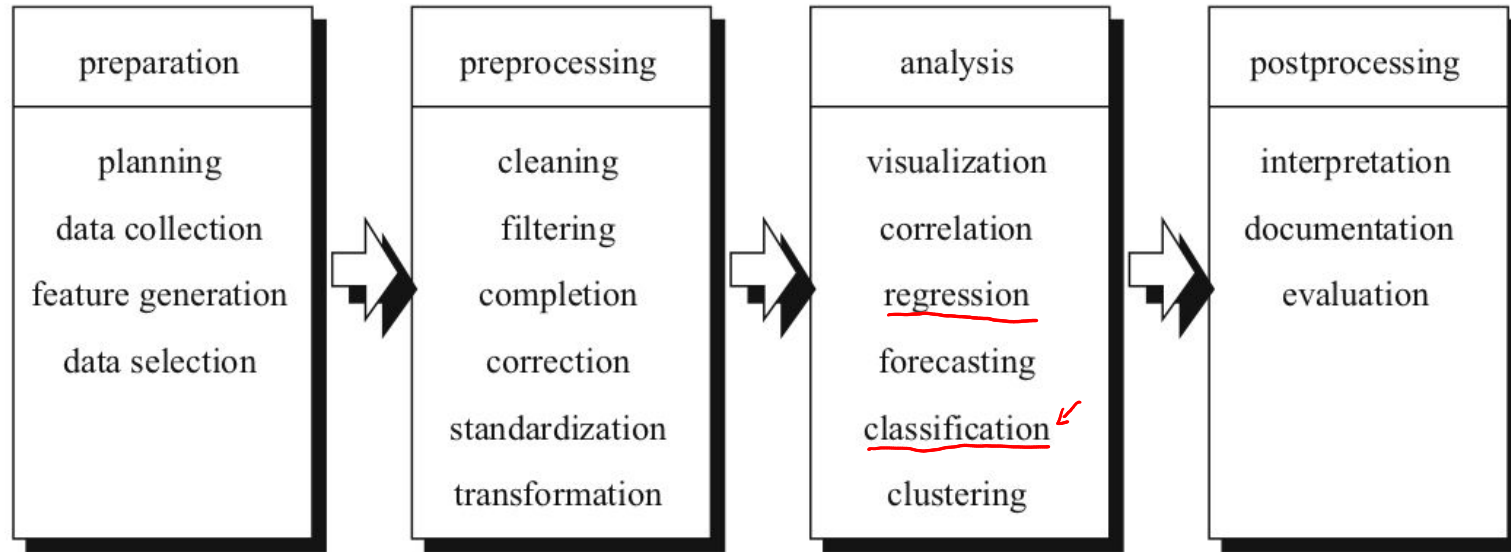


# Tarefas de mineração de séries temporais parte 2

# Etapas da descoberta de conhecimento



# Do ponto de vista de “data analytics”



**Fig. 1.1** Phases of data analysis projects

Classificação

# Aprendizado de máquina

“Machine Learning: the field of study that **gives computer the ability to learn** without being **explicitly programmed.**”

Arthur Samuel, 1959



# Definições de aprendizado de máquina

"A computer programming is said to **learn from experience** **E** with respect to some **task** **T** and some **performance measure** **P** if its performance on **T**, measured by **P**, improves with **E**."

Tom Mitchell, 1998



# Exemplo 1

"A computer programming is said to **learn from experience** **E** with respect to some **task** **T** and some **performance measure** **P** if its performance on **T**, measured by **P** improves with **E**."

Seu leitor de emails sabe toda vez que você marca um email como *spam*. Com isso, ele aprende a filtrar melhor essas mensagens. Perguntas: o que é **T**, nesse caso? **P**? **E**?

$$A_c = \frac{\# \text{acertos}}{\# \text{instâncias}} = \frac{\# \text{acertos}}{\# \text{emails}}$$

HAM






# Exemplo 2

Aprovação de crédito/empréstimo

Dados de cliente solicitando um novo empréstimo



Idade	25
Sexo	M
Salário	<u>R\$ 2.200,00</u>
Anos na residência atual	2,5
Anos no emprego atual	1
Saldo na conta	<u>R\$ 2,53</u>
...	

S ou N  $\Rightarrow$   
classif.

Qto  $\rightarrow$   
regressão

## Exemplo 2

O que compõe o aprendizado/decisão de cessão de crédito?

- Entrada:  $\mathcal{X}$
- Saída:  $\mathcal{Y}$
- Função:  $f: \mathcal{X} \rightarrow \mathcal{Y}$

## Exemplo 2

O que compõe o aprendizado/decisão de cessão de crédito?

- Entrada:  $x$  (aplicação / dados do cliente)
- Saída:  $y$  (bom pagador?) ✓
- Função:  $f: x \rightarrow y$  (função de aprovação de crédito) - função objetivo

# Exemplo 2

O que compõe o aprendizado/decisão de cessão de crédito?

- Entrada:  $x$
- Saída:  $y$
- Função:  $f: \mathcal{X} \rightarrow \mathcal{Y}$
- Dados:  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), \dots, (x_n, y_n)$

Dados históricos (**exemplos**)

ou instâncias

## Exemplo 2

O que compõe o aprendizado/decisão de cessão de crédito?

- Entrada:  $x$
- Saída:  $y$
- Função:  $f: \mathcal{X} \rightarrow \mathcal{Y}$
- Dados:  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), \dots, (x_n, y_n)$

- **Hipótese:**  $g: \underline{\mathcal{X}} \rightarrow \underline{\mathcal{Y}}$

# Exemplo 2

**Função desconhecida**

$$f: \mathcal{X} \rightarrow \mathcal{Y}$$

função de aprovação de crédito

**Exemplos de treinamento**

$$(x_1, y_1), \dots, (x_n, y_n)$$

dados históricos de clientes

**Hipótese**

$$g \approx f$$

Função aproximada para  
cessão de crédito

## Exemplo 2



# Premissa do aprendizado

Simplificando a premissa do aprendizado levantada por Tom Mitchell, aprendizado de máquina é a estimação de uma função de acordo com exemplos.

Premissa muito ampla → infinitas possibilidades



# Aprendizado supervisionado

"A computer programming is said to learn from experience  $E$  with respect to some task  $T$  and some performance measure  $P$  if its performance on  $T$ , measured by  $P$ , improves with  $E$ ."

possíveis  
anotações

No **aprendizado supervisionado**, temos alguém (ou algum processo) rotulando cada exemplo da base de experiência.

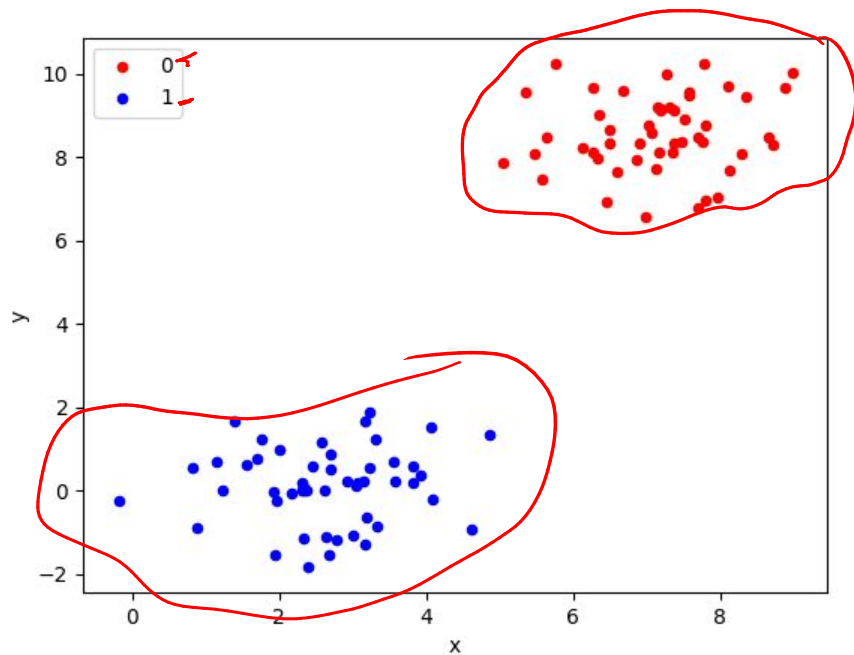
# Aprendizado supervisionado

Na tarefa de classificação, esses rótulos são **discretos**.

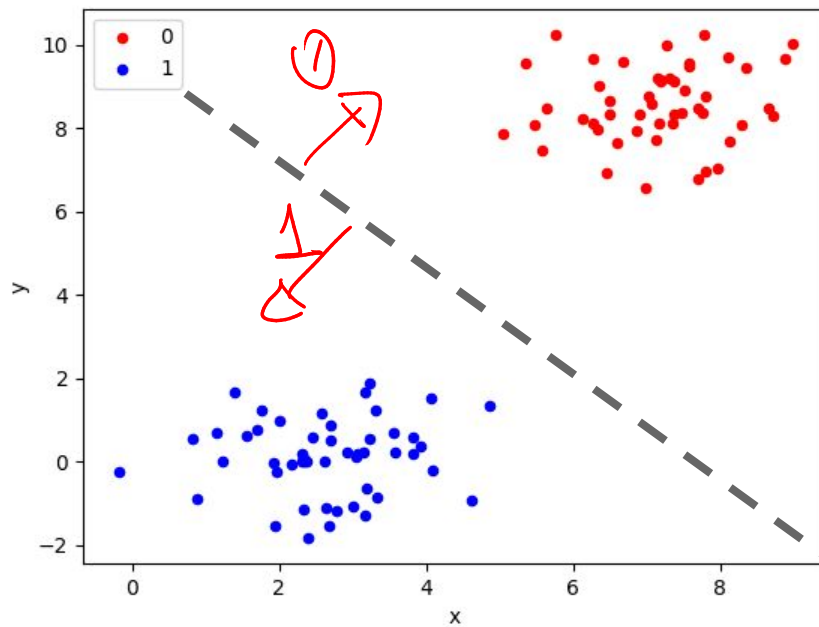
- Objetivo é, para um **dado não previamente rotulado**, descobrir a qual **classe** ele pertence

Lembrem-se do exemplo do empréstimo bancário.

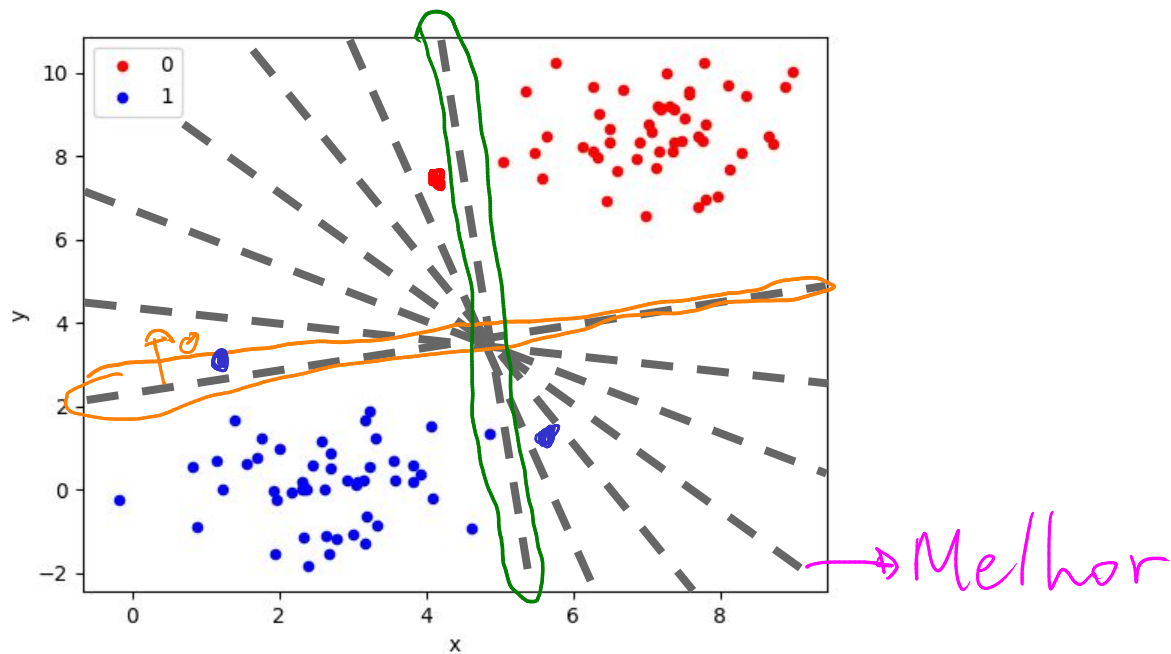
# Aprendizado supervisionado



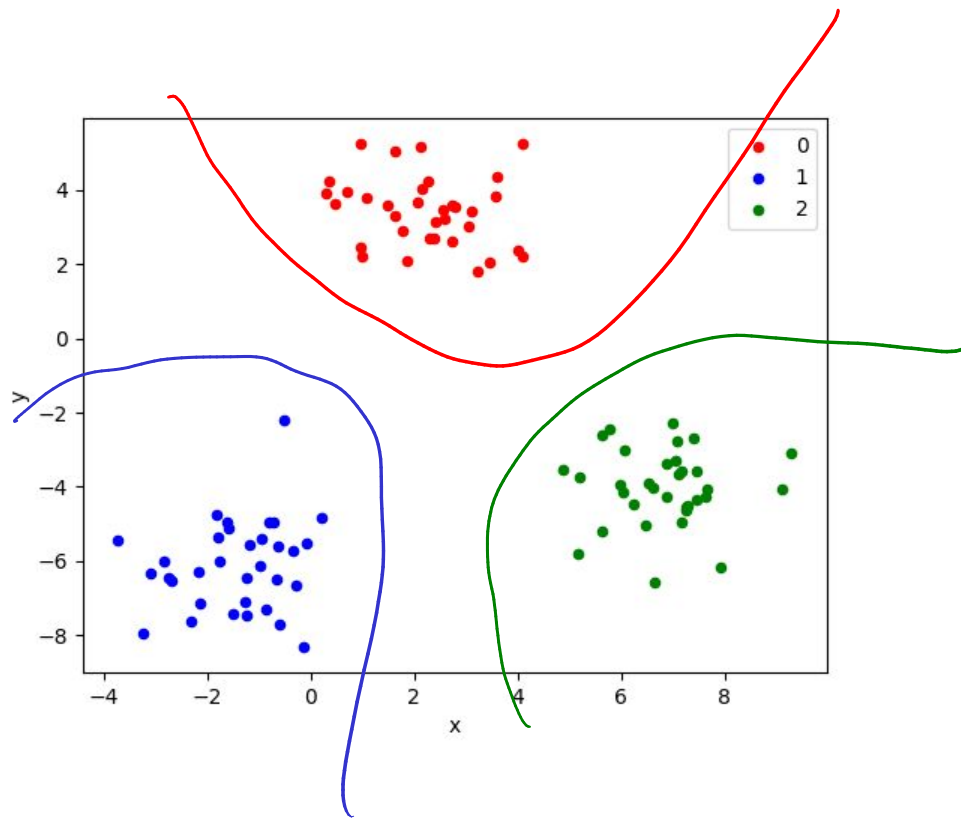
# Aprendizado supervisionado



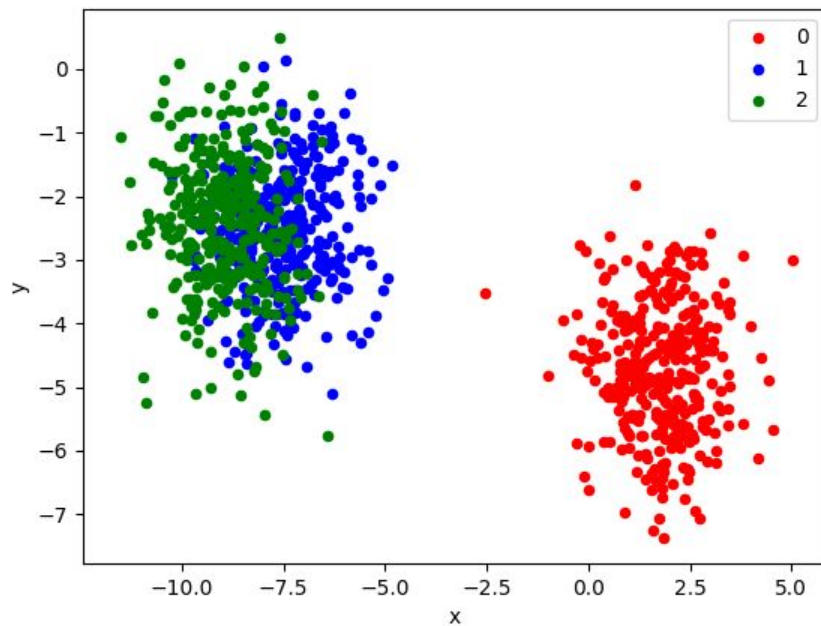
# Aprendizado supervisionado



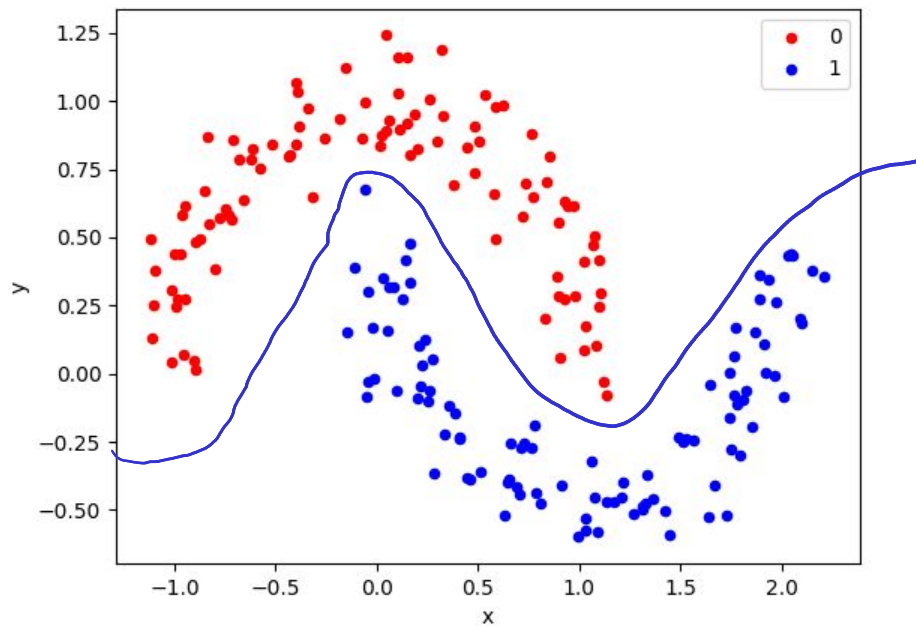
# Aprendizado supervisionado



# Aprendizado supervisionado



# Aprendizado supervisionado






# COLOCAR EXEMPLOS

COLOCAR EXEMPLOS

# Dados estruturados - tabulares



Tempo	Temperatura	Umidade	Vento	Joga
Chuvoso	71	91	Sim	Não
Ensolarado	69	70	Não	Sim
Ensolarado	80	90	Sim	Não
Nublado	83	86	Não	Sim
Chuvoso	70	96	Não	Sim
Chuvoso	65	70	Sim	Não
Nublado	64	65	Sim	Sim
Nublado	72	90	Sim	Sim
Ensolarado	75	70	Sim	Sim
Chuvoso	68	80	Não	Sim
Nublado	81	75	Não	Sim
Ensolarado	85	85	Não	Não
Ensolarado	72	95	Não	Não
Chuvoso	75	80	Não	Sim



LORENA, Ana Carolina; GAMA, João; FACELI, Katti. Inteligência Artificial: Uma abordagem de aprendizado de máquina. Grupo Gen-LTC, 2011.

# Dados estruturados - tabulares

Uma infinidade de possibilidades aqui

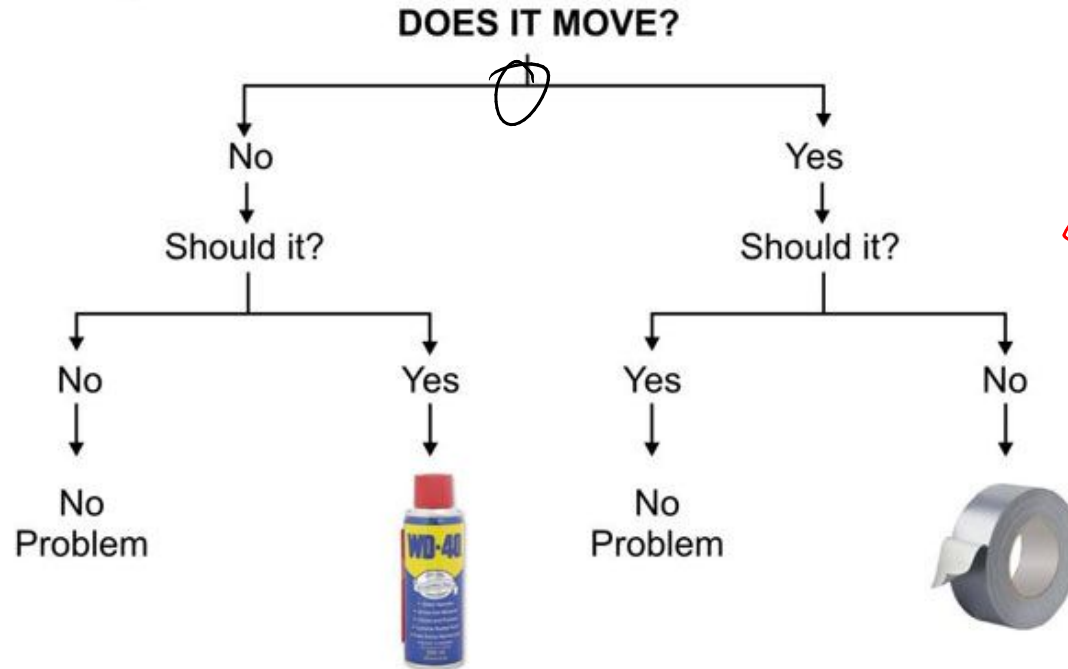
- Há cursos e mais cursos só de algoritmos de AM nesse cenário

Vamos ver um, mas só para ter uma ideia.

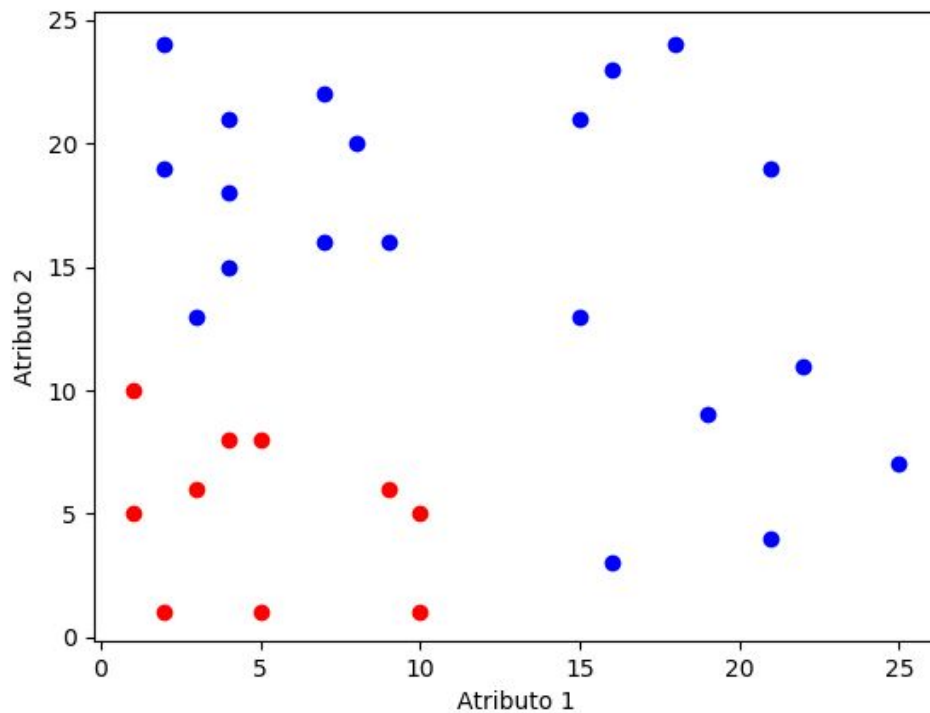
Depois, podemos usar o scikit-learn (ou orange, ou ....)

DATA MINING

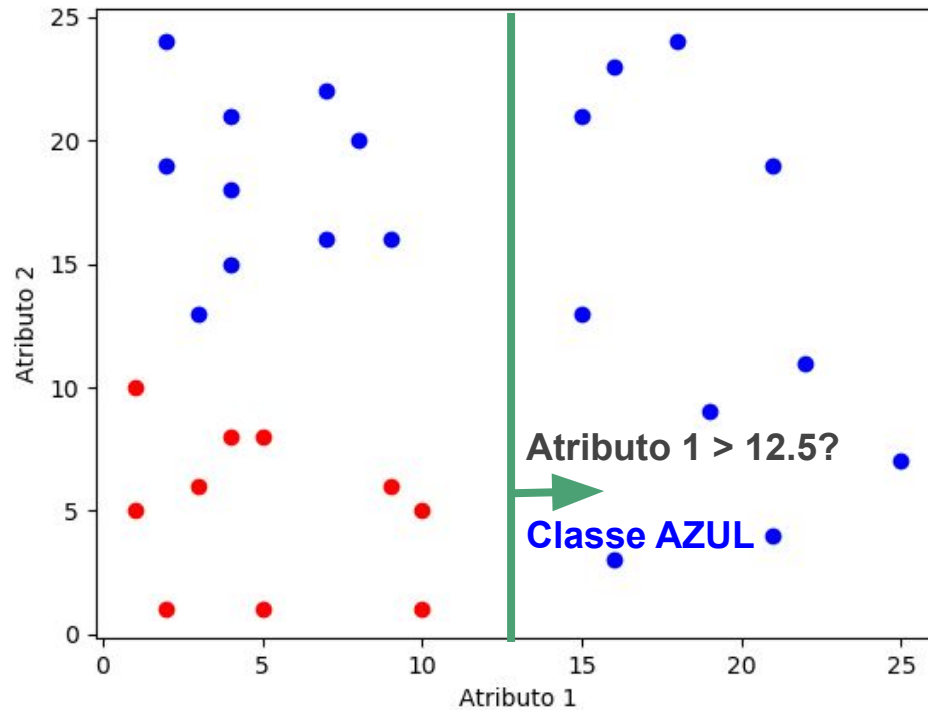
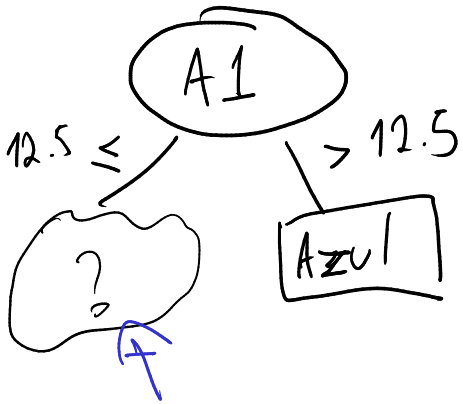
# Árvore de decisão



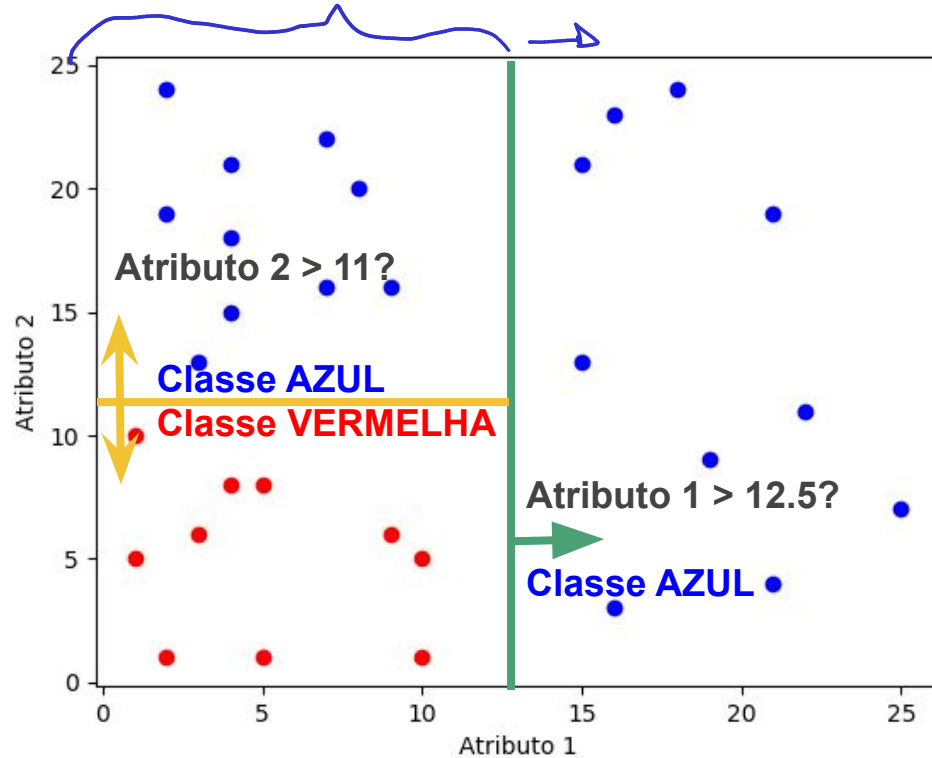
# Árvores de decisão



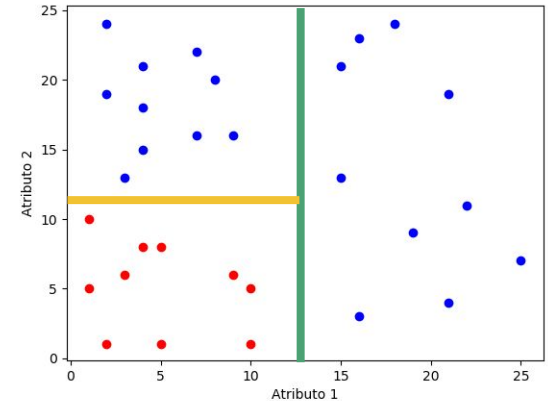
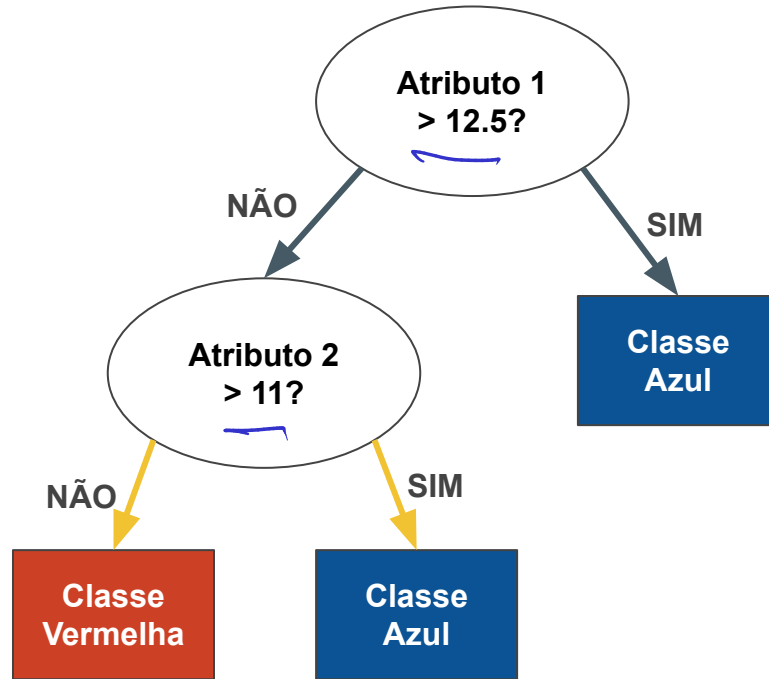
# Árvores de decisão



# Árvores de decisão

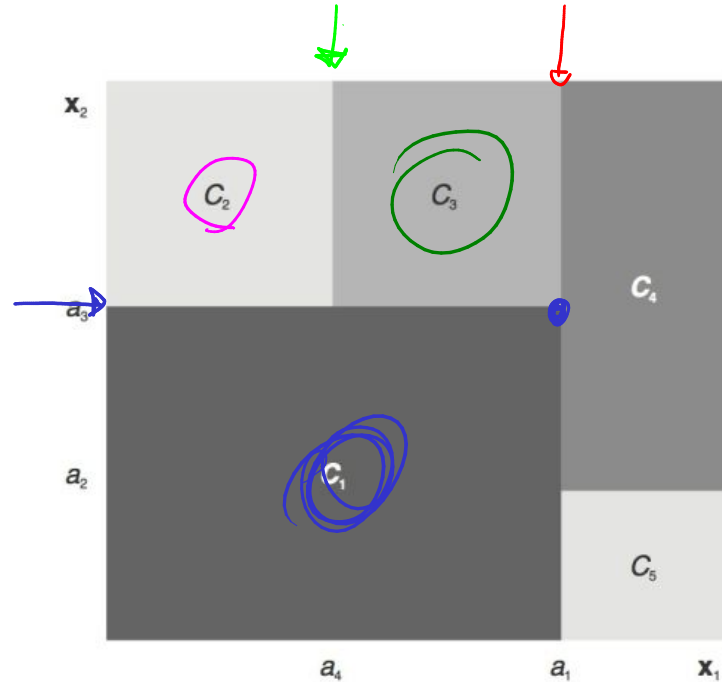
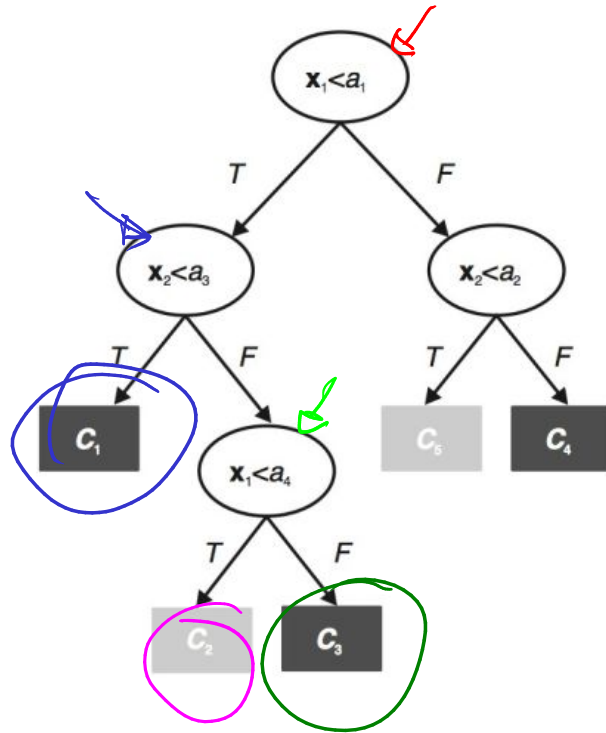


# Árvores de decisão





# Árvores de decisão



# Árvores de decisão

- Entropia  
- gini

---

**Entrada:** Um conjunto de treinamento  $D = \{(\mathbf{x}_i, y_i), i = 1, \dots, n\}$

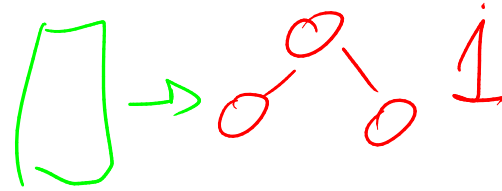
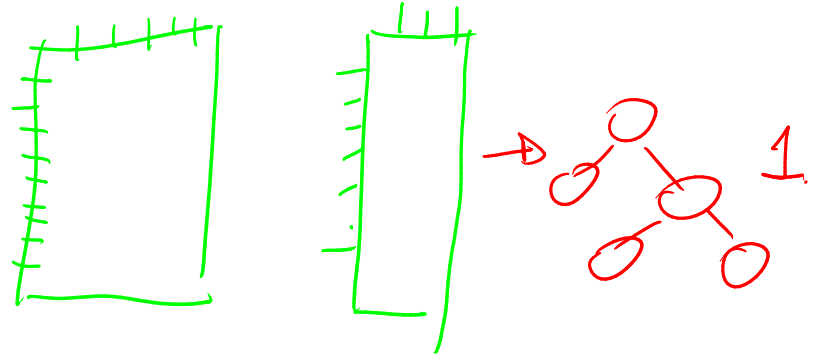
**Saída:** Árvore de Decisão

1 /\* **Função GeraÁrvore(D)** \*/ ;  
2 se *critério de parada(D) = Verdadeiro* então  
3     **Retorna:** um nó folha rotulado com a constante que minimiza a função perda ;  
4 fim  
5 Escolha o atributo que maximiza o critério de divisão em  $D$  ;  
6 para cada *partição dos exemplos  $D_i$  baseado nos valores do atributo escolhido* faça  
7     Induz uma subárvore  $Árvore_i = \text{GeraÁrvore}(D_i)$  ;  
8 fim  
9 **Retorna:** Árvore contendo um nó de decisão baseado no atributo escolhido, e descendentes  $Árvore_i$  ;

---

# Outros algoritmos

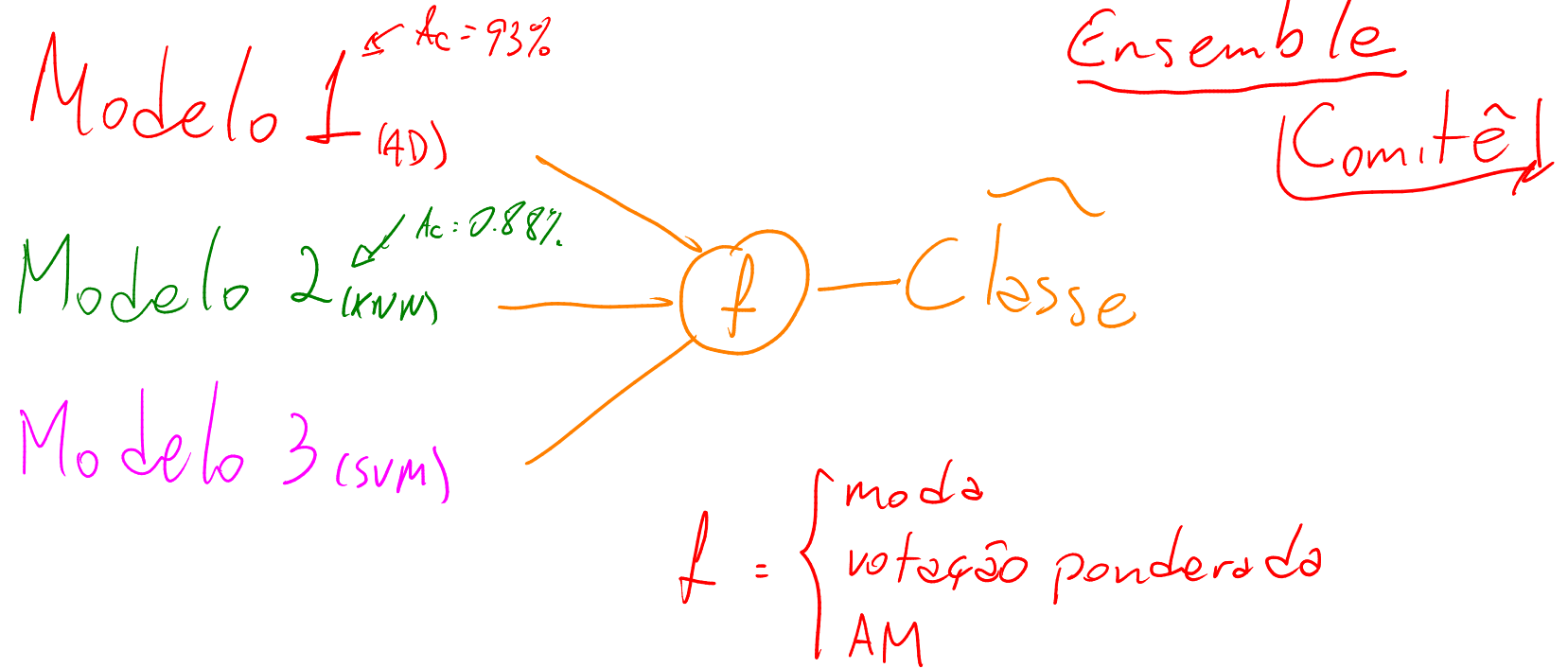
- Probabilísticos ↖
  - ↗ Naive Bayes, redes bayesianas...
- Baseados em similaridade ↖
  - k-vizinhos mais próximos (vamos com calma nesse, mais tarde)
- Baseados em otimização ↖
  - Regressão logística, SVM, redes neurais...
- Combinação de modelos *homogeneous*
  - Random forest, gradient boosting, ...



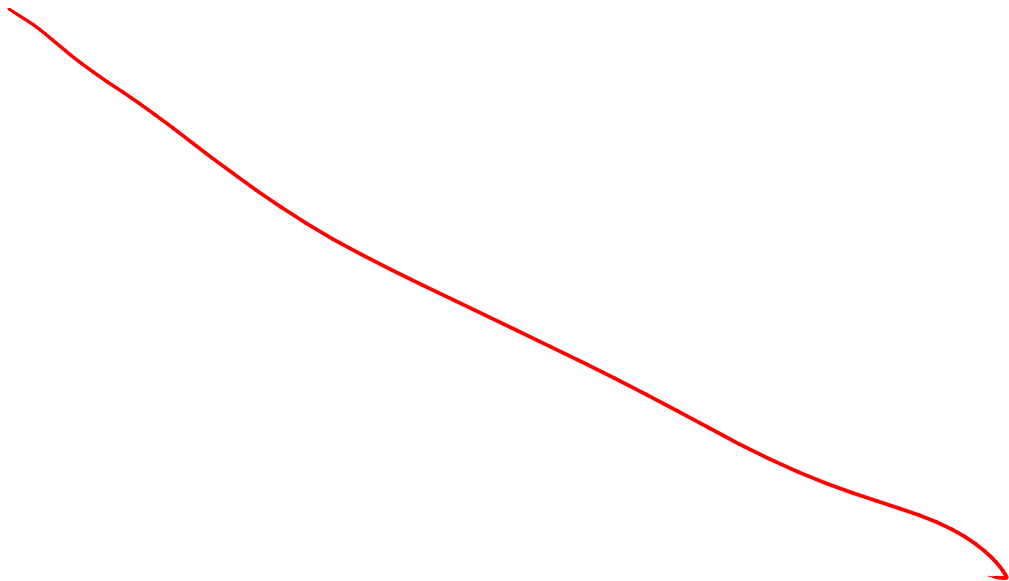
$$-\sum p \log(p)$$

0

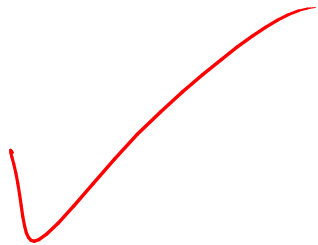
# Slide em branco para o prof divagar



Slide em branco para o prof divagar #2




Vamos ver na prática e já voltamos para séries



# Séries Temporais

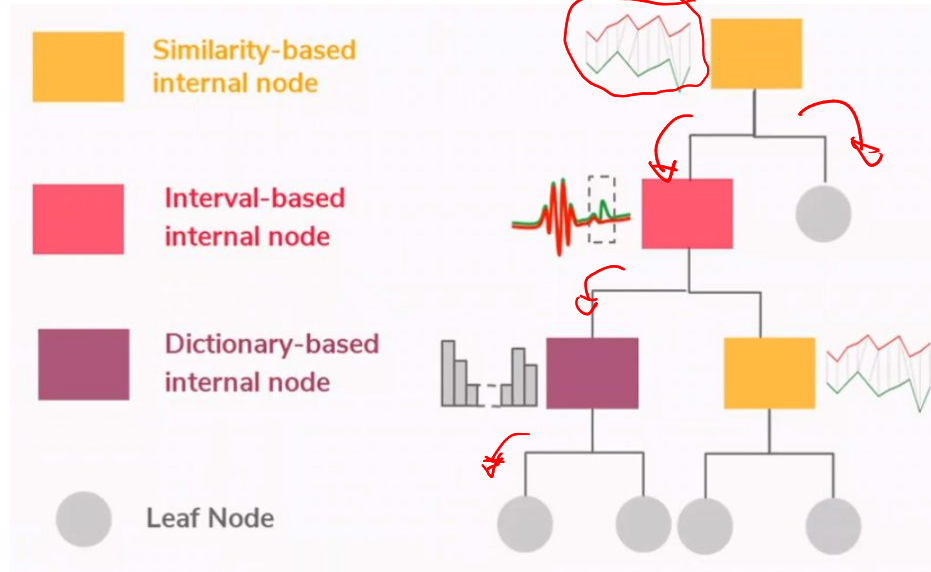
Podemos transformar as séries em dados estruturados 

- Chamamos essas técnicas de *transformers* ou extração de características
  - São coisas diferentes, mas são a mesma coisa (confuso)
- Vamos começar extraindo algumas características com tsfel 
  - Mas podemos usar qualquer outra: tsfresh, classes do sktime...

# Classificação de séries temporais

Há dezenas de algoritmos de TSC

- Alguns se aproveitam disso, como o TS-CHIEF



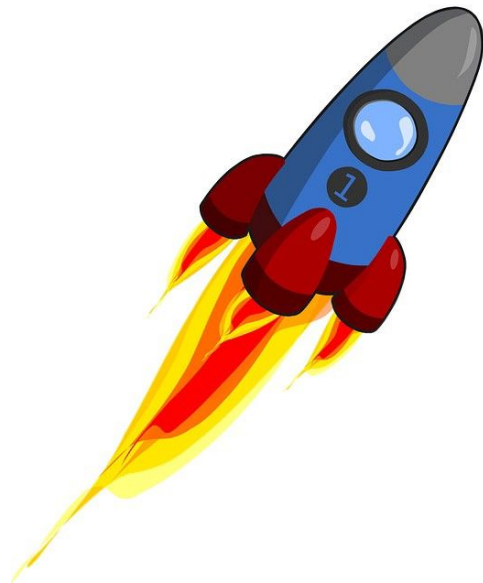
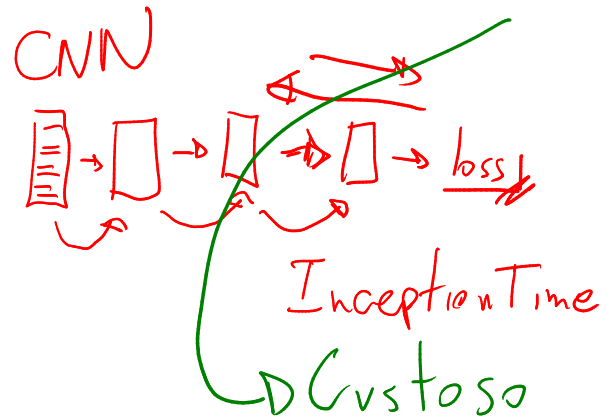
<https://youtu.be/SOnHXymw48k>



# Classificação de séries temporais

Há dezenas de algoritmos de TSC

- Muitos são baseados em estruturar os dados ←
- Um exemplo é o ROCKET
  - RandOm Convolutional KErnel Transform
  - <https://arxiv.org/pdf/1910.13051.pdf>
  - o ROCKET usa kernels convolucionais aleatórios
    - em outras palavras, filtros lineares com valores aleatórios
      - quase isso
    - Como cada kernel gera um “nova série”, é preciso agregar
      - Max e ppv

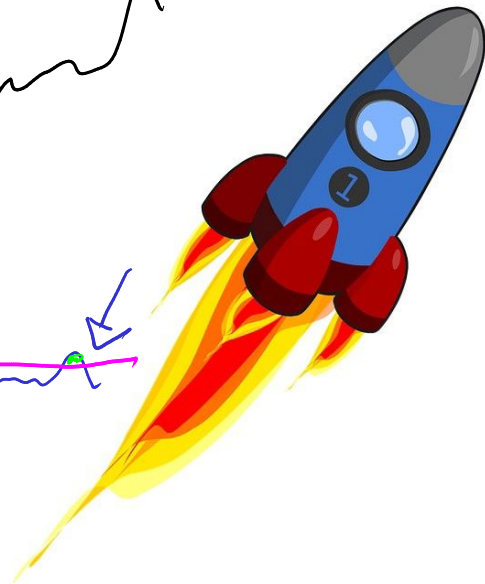
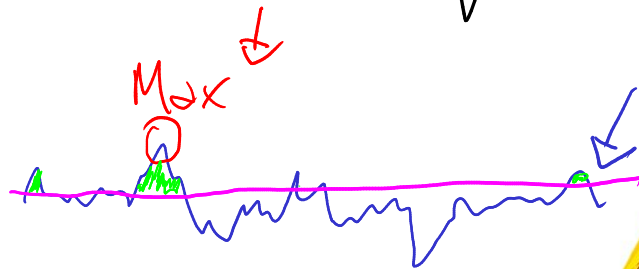
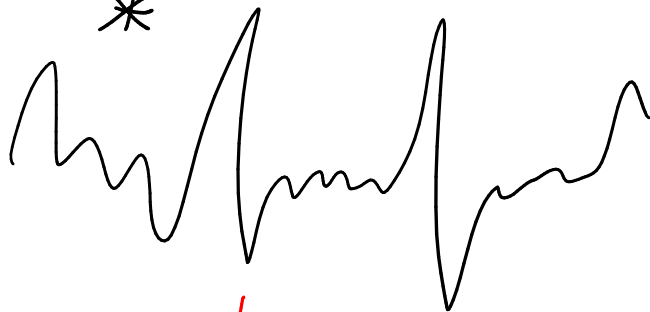


# Classificação de séries temporais

- Comprimento:  $\{7, 9, 11\}$
- Pesos:  $N(0,1)$
- Bias:  $(-1,1)$
- Dilatação:  $2^{U(0,A)}$
- Padding:  $\{T, F\}$

$\begin{bmatrix} .5 & .3 & .7 & .5 & .6 & .1 & .9 \end{bmatrix} + \text{Bias}$

\*



$R_{K1} \Rightarrow \max_{x_1}, PPV_1$

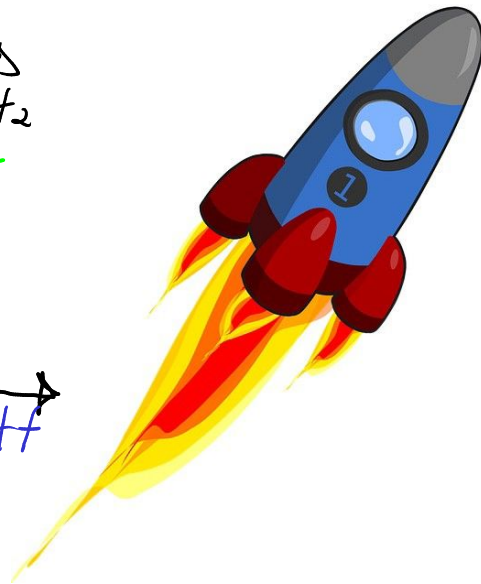
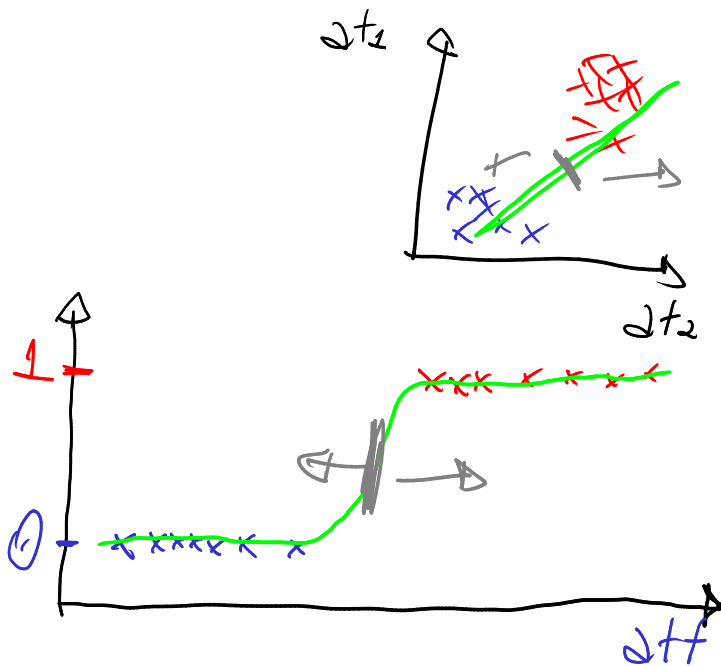
$R_{K2} \Rightarrow \max_{x_2}, PPV_2$



$m = 10K$

# Classificação de séries temporais

- Comprimento: {7, 9, 11}
- Pesos:  $N(0,1)$
- Bias:  $(-1,1)$
- Dilatação:  $2^{U(0,A)}$
- Padding: {T, F}



# Classificação de séries temporais

Prática

Dados em <http://timeseriesclassification.com/>

# Regressão extrínseca

# Regressão

**Tabela 3.13** *Exemplo de conjunto de dados para problema de regressão*

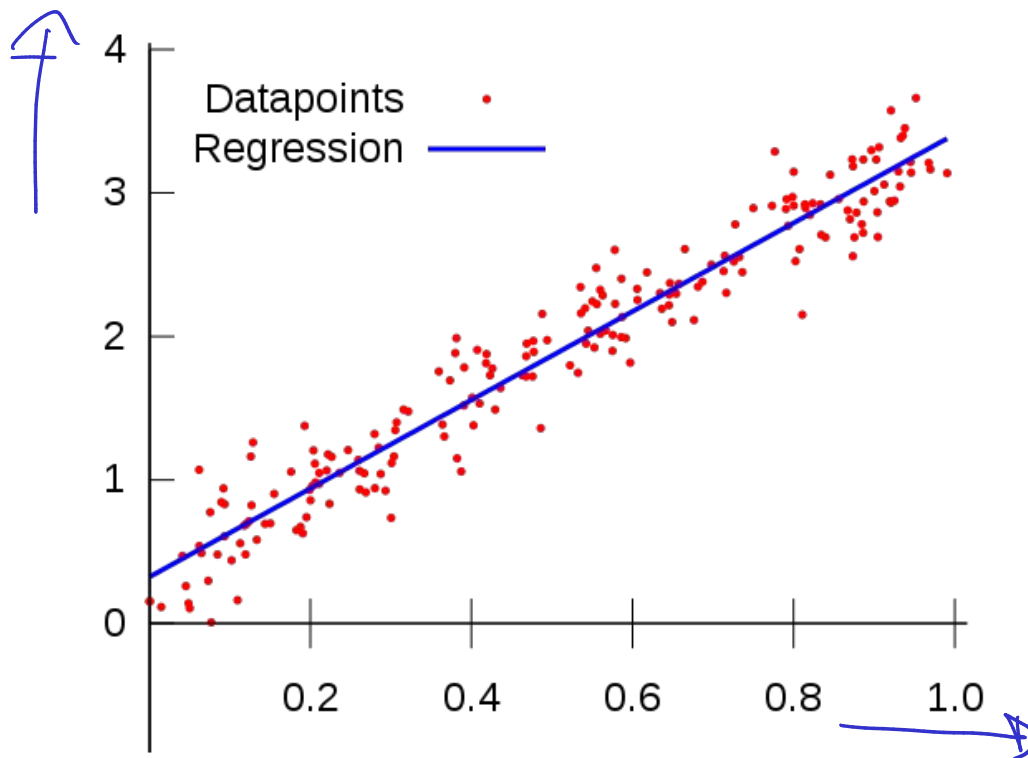
Fertilidade	Agricultura	Educação	Renda	Mortalidade
80,2	17,0	12	9,9	22,2
83,1	45,1	9	84,8	22,2
92,5	39,7	5	93,4	20,2
85,8	36,5	7	33,7	20,3
76,9	43,5	15	5,2	20,6

# Exemplos

- Qual é o valor comercial de uma casa, dadas as suas características e de seu bairro? ✓
- Quanto é o salário de um determinado profissional, dado o número de anos em que ele está na profissão? — *people analytics*
- Qual é o valor adequado para emprestar de dinheiro a um determinado cliente do banco? ✓
- Quanto uma determinada região da cidade produz de lixo de acordo com sua densidade populacional? ✓
- Dadas as características de uma rocha reservatório, quanto eu conseguirei extrair de petróleo? ✓



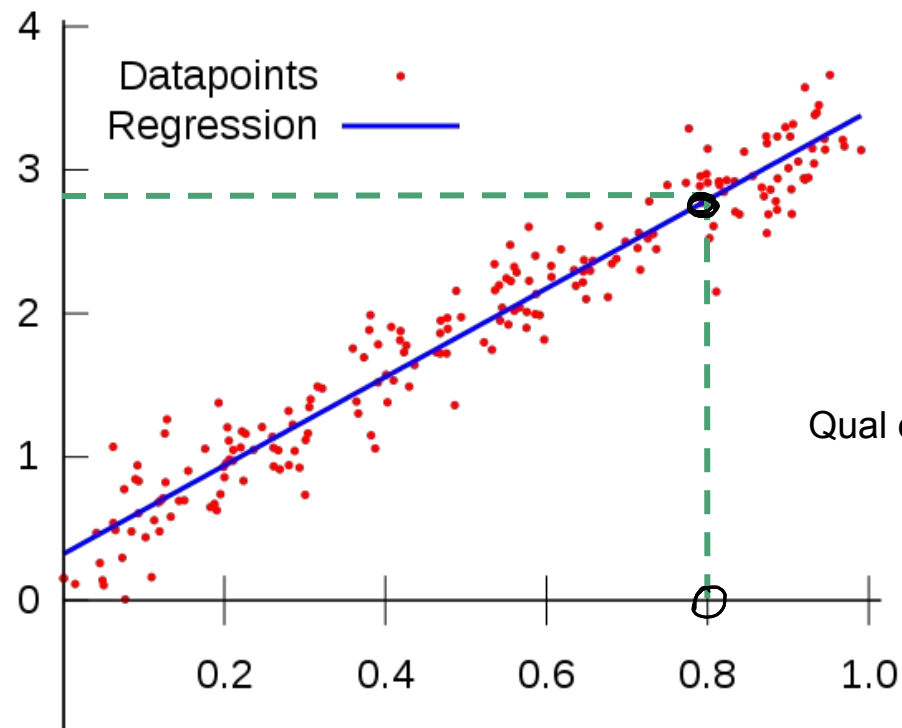
# Regressão linear



Fonte: [https://pt.wikipedia.org/wiki/Regress%C3%A3o\\_linear](https://pt.wikipedia.org/wiki/Regress%C3%A3o_linear)



# Regressão linear



2,9~

# Regressão não linear

Geralmente a complexidade do problema é muito maior

- Há versões de algoritmos de classificação de variados paradigmas para a tarefa de regressão (e vice-versa)



# Regressão extrínseca de séries temporais

Uma vez que o forecasting não deixa de ser uma regressão (AR), vamos chamar o problema abordado agora de *time series extrinsic regression* (TSER)

- O valor predito depende da série como um todo, não de observações recentes
- Faz-se um paralelo com classificação
  - Ex:
    - classificação em um ECG: patologia
    - TSER em um ECG: frequência cardíaca ou parâmetros clínicos

Geoff Webb

# Regressão extrínseca de séries temporais

Podemos atuar da mesma forma que em classificação

- Extrair features / transformar a série a fim de estruturar os dados
- Usar um regressor em vez de um classificador

Pronto, temos algoritmos para isso.

# Regressão extrínseca de séries temporais

Prática

Dados em <http://tseregression.org/>

